Aide à l'expertise des brevets par alignement avec les publications scientifiques

Kafil Hajlaoui*, **Pascal Cuxac***, **Jean Charles Lamirel****, **Claire François***
Kafil.Hajlaoui@inist.fr, Pascal.Cuxac@inist.fr, lamirel@loria.fr, Claire.Francois@inist.fr

(*) INIST CNRS, Vandœuvre-lès-Nancy, France
(**) INRIA team SYNALP-LORIA, Vandœuvre-lès-Nancy, France

Mots clefs :
Classification supervisée, Veille scientifique et technique, Brevets, K-PPV, Règles d'association
Keywords:
Supervised classification, Technological and scientific survey, Patents, KNN, Association rules
Palabras clave :
Clasificación supervisada, escudriñar científico y tecnológico, patentes, K-NN, reglas de asociación

Résumé

Ce travail s'inscrit dans le cadre du programme de recherche QUAERO[1], un vaste projet de recherche et d'innovation se rapportant au traitement automatique de contenus multimédias et multilingues. L'objectif abordé dans cet article est de proposer une méthode de classification automatique d'articles dans un plan de classement international de brevets relevant du même domaine. La finalité applicative de ce travail est de proposer une aide aux experts dans le processus d'évaluation de l'originalité et la nouveauté d'un brevet, en lui proposant les citations scientifiques les plus pertinentes. Ce problème soulève de nouveaux défis en catégorisation liés du fait que le plan de classement des brevets n'est pas directement adapté à la structure des documents scientifiques et que la répartition des exemples disponibles n'est pas nécessairement équilibrée entre les différentes classes d'apprentissage. Nous proposons pour les résoudre d'appliquer une amélioration de l'algorithme des K-plus-proches-voisins (K-PPV) se basant sur l'exploitation des règles d'associations entre les termes descripteurs des documents et ceux des classes de brevets. En utilisant conjointement comme référentiels une base de brevets du domaine de la pharmacologie et une base bibliographique du même domaine issue de la collection Medline, nous montrons que cette nouvelle technique de catégorisation, qui combine les avantages des approches numériques et ceux des approches symboliques, permet d'améliorer sensiblement les performances de catégorisation, relativement aux méthodes de catégorisation usuelles, dans le cas du problème posé.

[1] http://www.quaero.org

1. Introduction

La catégorisation automatique de textes (CAT) vise à regrouper, souvent selon des thèmes communs, les documents ayant des caractéristiques spécifiques et homogènes [3]. La première étape ce type de catégorisation est la transformation des documents en une représentation appropriée pour le classifieur. Cette transformation vise à pondérer et à réduire l'espace de représentation des documents tout en ménageant la possibilité de discriminer entre ces derniers. Elle comprend usuellement des opérations de suppression des mots vides, de lemmatisation, de sélection et de pondération des descripteurs. La deuxième étape est l'apprentissage : le système apprend à classer les documents selon un modèle de classement où les classes sont prédéterminées et les exemples sont connus et correctement étiquetés d'avance.

La catégorisation automatique de textes a été l'un des domaines les plus étudiés en apprentissage automatique [8]. En conséquence, une variété d'algorithmes de classification ont été développés et évalués, souvent dans des applications telles que le filtrage des mails [4] ou l'analyse des opinions et des sentiments [18]. Dans le domaine des sciences sociales, l'apprentissage automatique a été utilisé dans la classification d'actualités [19] [6], ou des blogues [5]. Parmi les méthodes d'apprentissage les plus souvent utilisées, figurent les réseaux de neurones [29] [23], les K-plus-proches-voisins (K-PPV) [30], les arbres de décision [15] [27] [2], les réseaux bayésiens [16] [10], les machines à vecteurs support (SVM) [11], et plus récemment, les méthodes basées sur le boosting [22] [9]. Bien que beaucoup de méthodes développées dans le domaine de la catégorisation automatique de textes ont permis d'atteindre des niveaux de précision appréciables lorsqu'il s'agit de textes à structure simple (par ex. courriels, résumés, etc.), il reste néanmoins encore de nombreux défis à relever dans le cas de documents complexes, ou, comme le cas que nous traitons, si le plan de classement des documents n'est pas directement adapté à leur contenu et si la répartition des exemples entre les différentes classes d'apprentissage n'est pas équilibrée.

Plusieurs travaux ont été réalisés plus spécifiquement sur des données issues de la base Medline. Ces travaux illustrent plus particulièrement l'importance des étapes de prétraitement et de représentation des données dans le cadre de la catégorisation des textes. Dans [14], les auteurs montrent qu'avec une représentation de textes basée sur l'approche dite « sac de mots », la pondération des termes extraits augmente significativement la performance du classifieur. Pour classer un article scientifique dans un sujet (thème), Suomela et Andrade [26], se basent quant à eux sur la fréquence des termes, en restreignant ces derniers à des classes lexicales prédéfinies (Noms, Adjectifs, Verbes). Les auteurs évaluent leur proposition en utilisant des thèmes issus de la base Medline, et obtiennent une performance F-score de 65%. La même approche est reprise par le système MedlineRanker web-service [7] qui permet de retrouver une liste pertinente de notices Medline à partir d'un ensemble de mots-clés définis par l'utilisateur. Les travaux de Yin et al. [32] portent sur l'identification et l'extraction des interactions entre protéines à partir des articles Medline. Les documents sont traités en utilisant des bi-grammes. Avec la méthode SVM, les auteurs obtiennent une performance de 50% de vrais positifs, et un taux de rappel de 51%. Récemment la tâche d'évaluation Bio-creative III a proposé comme challenge la classification d'articles Medline spécifiques au domaine biomédical [13]. Sur cette collection, les meilleures performances ont été obtenues, avec une précision de 89,2% et un F-score de 61,3%.

L'évaluation des brevets est une opération jusqu'ici manuelle qui fait intervenir des groupes d'experts ayant des compétences dans le domaine d'analyse et qui connaissent parfaitement l'objet des brevets. Elle s'appuie sur des références et des citations vers des documents scientifiques pertinents (articles, thèses, ouvrages…). Un classement automatisé des publications dans les classes de brevets peut donc constituer une aide précieuse pour les experts. Cette démarche implique de classer des articles scientifiques (notices) dans un plan de classement des brevets ; il ne s'agit donc pas d'une problématique traditionnelle de

la classification automatique, car le plan de la classification utilisé n'est pas a priori adapté à une classification de notices bibliographiques d'articles scientifiques.

Dans ce nouveau contexte, deux alternatives sont possibles. Une première alternative est de concevoir une passerelle entre le plan de classement des publications et celui des brevets. Cette démarche est cependant difficile à mettre en œuvre car elle implique l'exploitation intensive de techniques très lourdes de comparaison d'arbres (matérialisés ici par les plans de classement), et doit s'opérer en partie de manière supervisée. Une deuxième alternative est d'élaborer un système de classification des notices bibliographiques dans le plan des brevets. Elle est basée sur l'hypothèse que les citations scientifiques qui apparaissent dans un brevet sont fortement liées au domaine du brevet, donc au code de classement de ce dernier. Dans ce cadre, le corpus d'apprentissage d'une classe donnée représentera alors l'ensemble des citations extraites des brevets de cette classe. Même si cette idée est plus facile à mettre en œuvre, elle implique néanmoins de résoudre un problème supplémentaire qui est celui d'avoir à disposition un nombre équivalent d'exemple d'apprentissage (i.e. de citations de publications) dans chacune des classes de brevets, ces classes n'ayant pas nécessairement elles-mêmes un effectif homogène, en termes de brevets.

Dans les sections suivantes, nous menons une expérimentation complète de catégorisation des publications à partir d'un corpus de brevets issus du domaine de la pharmacologie et d'un corpus bibliographique issu de la collection Medline. Dans la première section, nous présentons notre stratégie de constitution du corpus expérimental et nous illustrons les phénomènes de déséquilibre des exemples d'apprentissage et de similarité des classes qu'il est possible d'observer. En exploitant les méthodes de catégorisation usuelles, nous illustrons ensuite, dans la seconde section, l'influence de la stratégie de choix des termes descripteurs utilisés pour les documents-exemples sur les résultats de catégorisation. Deux approches sont plus particulièrement abordées, la première basée sur l'exploitation directe des mots-clés Medline, la seconde basée sur l'extraction d'index à partir du texte plein des titres et résumés en utilisant une plate-forme de traitement linguistique. Dans la troisième section, nous présentons une adaptation de l'algorithme des K-plus-proches-voisins (K-PPV) se basant sur l'exploitation des règles d'associations identifiées entre les termes descripteurs des documents et ceux des classes de brevets. Nous montrons qu'elle permet d'améliorer les résultats obtenus dans notre contexte d'apprentissage. La dernière section présente notre conclusion et nos perspectives.

2. Constitution et indexation du corpus

2.1 Extraction des données

La ressource principale de notre corpus est une collection de brevets du domaine de la pharmacologie auxquels sont associés des citations bibliographiques. Les notices brevet au format XML sont au nombre de 6387 réparties dans les 15 classes de la catégorie A61K (préparations à usage médical…). Comme l'illustre la figure 1, nous commençons par l'extraction des références à partir des brevets. Grace à des robots web, nous interrogeons une base de données de publications pour extraire les notices relatives aux références collectées. Chaque notice est ensuite étiquetée par la classe du brevet citant. L'ensemble des notices étiquetées représente notre corpus d'apprentissage.

Nous avons interrogé la base de données Medline[2] qui est spécialisée dans le domaine de la médecine et qui, d'autre part, bénéficie de mises à jour régulières. A partir des 6387 brevets, nous avons extraits 25887 références de types bases de données, livres, encyclopédie… et articles scientifiques. L'interrogation de Medline avec les références de types articles scientifiques nous a fourni 7501 notices, ce qui représente un rappel de 90% relatif à ce type de références.

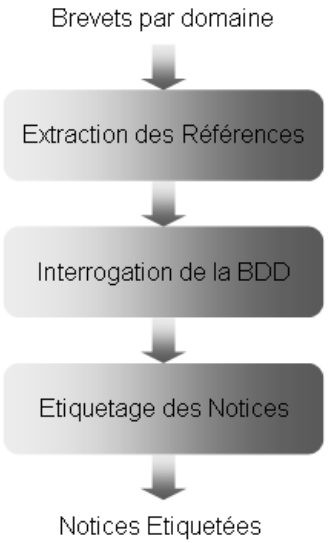

Figure 1 : Processus de construction du corpus d'apprentissage

La figure 2 résume la répartition des notices du corpus. Au vu de ces chiffres, il semble clair qu'un des critères importants pour le choix de la méthode de classification sera sans doute sa capacité à traiter le déséquilibre des exemples entre les classes. En effet la distribution des notices entre les classes est très hétérogène : nous avons des sous-classes qui ne contiennent que quelques dizaines de notices en comparaison avec d'autres qui en contiennent plus de 2500.

[2] http://www.ncbi.nlm.nih.gov/pubmed/

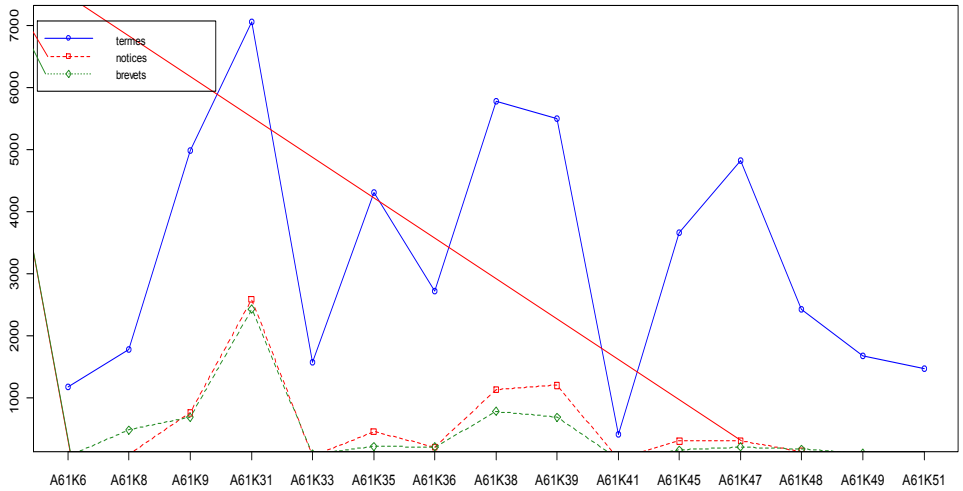

Figure 2 : Répartition des données dans les classes : brevets-notices-termes

2.2 Représentation des données

Dans la classification automatique de textes, le choix de représentation des documents est une étape cruciale. Une approche fréquente consiste à faire appel à une représentation dite « sac de mots », où la seule information utilisée est la présence et/ou la fréquence de certains mots. Dans notre contexte, nous utilisons une représentation vectorielle des documents selon le modèle de Salton [21], de manière à pouvoir représenter les documents sous forme de vecteurs de mots pondérés. Chaque notice de la collection est représentée comme un vecteur dans un espace à N dimensions, où N est le nombre total des termes extraits de la collection de notices. L'ensemble de la collection des notices est représenté par une matrice de dimension (N +1) x J, où J est le nombre de notices dans la collection. Chaque ligne j de cette matrice est un vecteur à N dimensions auquel on ajoute l'étiquette de la classe pour la notice j. Si un descripteur i n'est pas produit par la notice j, alors la valeur a_{ij} de la matrice vaut 0. Dans le cas contraire, a_{ij} prend une valeur positive. La méthode pour calculer cette valeur positive dépend à la fois du choix de représentation et du choix de pondération des descripteurs.

Nous avons ensuite construit nos descriptions à partir de deux approches différentes, une première basée sur les mots-clés présents dans les notices, et une approche alternative, basée sur les lemmes issues du traitement du texte plein des résumés à partir de méthodes de traitement automatique des langues (TAL). L'objectif de cette dernière approche est d'améliorer la représentation du contenu des documents. Pour ce faire nous utilisons le programme TreeTagger [24] qui est à la fois un étiqueteur et un lemmatiseur développé par l'Institute for Computational Linguistics de l'Université de Stuttgart. Un étiqueteur est un outil

qui permet d'annoter automatiquement un texte avec des informations morphosyntaxiques alors qu'un lemmatiseur associe un lemme, ou racine syntaxique, à chaque mot du texte. Dans un premier temps, les documents ont été lemmatisés. La suite des analyses a été effectuée sur les formes lemmatisées, sauf lorsque le mot était inconnu du tagger et, dans ce cas, la forme originale a été conservée. Les signes de ponctuation et les nombres, identifiés par le tagger, ont été supprimés. Un exemple de sortie du programme TreeTagger[3] est donné à la figure 3.

The	DT	the
most	RBS	most
widely	RB	widely
used	VVN	use
therapeutic	JJ	therapeutic
modality	NN	modality
is	VBZ	be
chemical	JJ	chemical
pleurodesis	**NN**	**<unknown>**

Figure 3 : Exemple d'une phrase étiquetée et lemmatisée par TreeTagger[3]

La sélection d'attributs selon les catégories grammaticales permet d'identifier par exemple des traits de jugement subjectif pour la classification de documents par genre ou par opinion. Il serait donc pertinent, dans notre cas, de mesurer l'impact de l'utilisation de descripteurs fondés sur la sélection de catégorie(s) grammaticale(s). Cette étude peut permettre de réduire de manière conséquente la taille de l'espace de description. De plus, nous avons décidé de retenir les mots lemmatisés '<unknown>' par TreeTagger et catégorisés comme nom sous leur forme non lemmatisée (NN). Par exemple, dans le cas illustré à la figure 3, nous avons retenu le mot 'pleurodesis' pour la classification.

La pondération fréquentielle se fonde sur le nombre d'occurrences des descripteurs dans un document. Cependant, en procédant de la sorte, on donne une importance trop grande aux descripteurs qui apparaissent très souvent dans un grand nombre de documents et qui sont peu représentatifs d'un document en particulier. On trouve dans la littérature [28] [20] [12] une autre mesure de poids, connue sous le nom de TF.IDF (Term Frequency Inverse Document Frequency). Elle permet de mesurer l'importance du mot en fonction de sa fréquence dans le document (TF=Term Frequency) pondérée par la fréquence d'apparition du terme dans tout le corpus (IDF=Inverse Document Frequency).

$$Tf.Idf(t_k, D_j) = TF(t_k, D_j) \times Idf(t_k)$$

où $TF(t_k, D_j)$ est le nombre d'occurrences de t_k dans D_j, et,

$$Idf(t_k) = \frac{\log|S|}{DF(t_k)}$$

[3] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/Penn-Treebank-Tagset.pdf

où $|S|$ est le nombre de documents dans le corpus et $DF(t_k)$ est le nombre de documents contenant t_k .

Cette dernière mesure permet de donner un poids plus important aux mots discriminants d'un document. Inversement, un terme apparaissant dans tous les documents du corpus aura un poids faible, voire nul.

Dans l'ensemble des tests, nous appliquons deux techniques de pondération différentes selon les descripteurs extraits. Pour les lemmes nous pondérons conjointement par la technique fréquentielle (TF) normalisée par la valeur maximum de fréquence et par la technique IDF. Pour les mots-clés, la technique TF n'a pas de sens puisque les termes d'indexation documentalistes ne sont pas redondants. C'est pourquoi nous n'utilisons dans ce cas que la technique IDF pour la pondération.

3. Classification

Pour évaluer la pertinence des différentes méthodes d'indexation et de pondération, nous avons choisi d'utiliser trois classifieurs pour la classification supervisée : un classifieur de type K-plus-proches-voisins (K-PPV) exploitant une distance euclidienne, un classifieur fondé sur les machines à vecteurs supports (SVM) et un classifieur probabiliste (Naive Bayes). Le choix s'est fixé sur ces trois méthodes parce qu'il s'agit des algorithmes d'apprentissage supervisé qui donnent le plus souvent les meilleurs résultats pour la classification des textes [25] [31]. Ces algorithmes sont utilisables sous l'environnement Weka[4].

Dans la le cas des de l'indexation basée sur les lemmes, nous présentons les différentes expérimentations que nous avons réalisées, en faisant varier les catégories grammaticales prises en compte dans l'indexation (A : Adjectif, N : Nom, NA : Nom+Adjectif, NV : Nom+Verbe, VA : Verbe+Adjectif, NVA : Nom+Verbe+Adjectif). Les résultats de la classification sont présentés en termes de précision et de rappel. Une précision de 100% signifie que toutes les notices sont classées dans la bonne catégorie. Cette mesure est calculée après l'application d'une validation croisée en dix sous-ensembles (90% du corpus est utilisée pour l'apprentissage et 10% pour le test). Le rappel est le pourcentage de réponses correctes qui sont données.

Tableau 1 : Résultat de la classification basée sur l'indexation par mots-clés

Algorithmes	KNN				NB				SVM			
mesure	Booléen		IDF		Booléen		IDF		Booléen		IDF	
	Précision	Rappel	Précision	Rappel	Précision	Rappel	Précision	Rappel	Précision	Rappel	Précision	Rappel
mots-clés	*0,39*	*0,39*	*0,39*	0,43	0,4	**0,47**	**0,43**	0,44	0,4	0,45	0,4	0,45

[4] http://www.cs.waikato.ac.nz/ml/weka/index.html

Tableau 2 : Résultat de la classification basée sur l'indexation par lemmes

Algorithmes	KNN				NB				SVM			
mesure	Fréquentiel		TF-IDF		Fréquentiel		TF-IDF		Fréquentiel		TF-IDF	
	Précision	Rappel	Précision	Rappel	Précision	Rappel	Précision	Rappel	Précision	Rappel	Précision	Rappel
A	0,42	*0,36*	0,42	0,36	0,38	0,2	0,37	*0,18*	0,45	0,46	0,45	0,46
N	0,5	0,41	0,52	0,4	0,43	0,31	0,44	0,28	0,54	0,55	0,54	0,55
NA	0,55	0,4	0,57	0,39	0,45	0,36	0,46	0,36	0,55	0,55	0,55	0,55
NV	0,49	0,38	0,52	0,38	0,44	0,35	0,44	0,31	0,53	0,54	0,53	0,54
NVA	0,6	0,54	**0,61**	**0,55**	0,44	0,34	0,45	0,34	0,54	0,55	**0,55**	**0,55**

Les tableaux 1 et 2 montrent la précision et le rappel obtenus pour la classification avec les trois algorithmes d'apprentissage sur le même corpus de notices bibliographiques, en faisant varier les méthodes d'indexation. Ils permettent de montrer à la fois que les approches utilisées pour la classification, les méthodes d'indexation et les méthodes de pondération des descripteurs ne sont pas équivalentes dans le cas du problème posé. Ainsi, les meilleurs résultats sur notre corpus sont obtenus avec la méthode K-PPV, combinée à une indexation basée sur les lemmes, impliquant les trois catégories grammaticales (Noms, Verbes, Adjectifs), et une pondération de type TF-IDF, avec une Précision de 61% et un Rappel de 55%. Ces résultats peuvent cependant être considérés comme moyens. Ceci peut s'expliquer par le fait que les exemples d'apprentissage ne sont pas équitablement répartis entre les classes (cf. figure 2), mais également que les classes sont très proches les uns des autres. Une similarité classe/classe a été calculée et elle montre bien cette proximité (figure 4), rendant difficile pour tout modèle la détection exacte de la bonne classe. La figure 4 montre ainsi que plus de 70% des couples de classes ont une similarité entre 0,5 et 0,9.

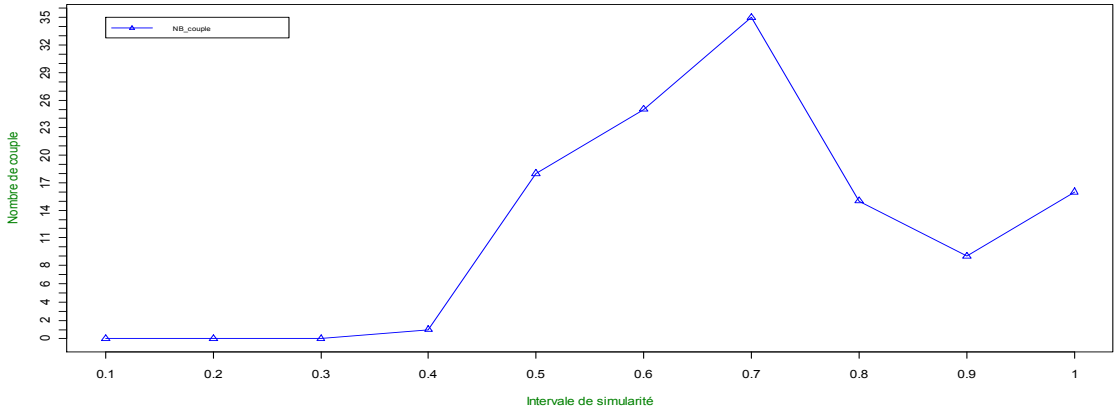

Figure 4 : Similarité Classe/Classe

Nous proposons donc à la section suivante une amélioration basée sur la meilleure méthode, à savoir celle des K-PPV, et susceptibles de prendre en compte les caractéristiques spécifiques du corpus, à savoir le déséquilibre entre les classes d'apprentissage et la forte similarité entre ces dernières.

4. La méthode K-PPVBA-2T

Dans cette partie, nous nous intéressons à une amélioration de l'algorithme K-PPV exploitable dans le contexte de notre problème. Nous allons présenter une définition générale des règles d'association. Ensuite, nous suggérerons une nouvelle approche pour calculer le poids des attributs des classes par l'utilisation d'un type particulier de règles d'association. Enfin, pour obtenir plus de précision dans la classification des données nous présenterons un nouvel algorithme appelé « K-PPVBA-2T » inspiré de la méthode développée antérieurement par Mordian et al. [17].

4.1. Règles d'association

La méthode d'extraction de règles d'associations représente une méthode permettant de découvrir des relations pertinentes entre deux ou plusieurs variables. Cette méthode se base sur des lois locales et ne nécessite pas d'intervention de l'utilisateur (on laisse le système s'auto-organiser). Elle permet d'identifier, à partir d'un ensemble de transactions, un ensemble de règles qui expriment une possibilité d'association entre différents items (mots, attributs, concepts). Une transaction est une succession d'items exprimés selon un ordre donné ; de plus, les transactions peuvent être de longueurs différentes. La pertinence d'une règle d'association ainsi extraite est mesurée par son indice de support et son indice de confiance. Si on a une règle d'association : $X \rightarrow Y$ alors les indices de support et confiance sont définies par les deux équations suivantes :

$$Support = P(X \cup Y),\ Confiance = P(Y|X)$$

où $P(X \cup Y)$ indique la probabilité qu'un enregistrement contienne à la fois X et Y, et $P(Y|X)$ est la probabilité conditionnelle d'avoir Y sachant qu'on a X.

La première méthode efficace d'extraction de règles d'association a été introduite par Agrawal pour l'analyse du panier de la ménagère, par l'intermédiaire de l'algorithme Apriori [AGR]. Le fonctionnement de cet algorithme peut être décomposé en deux phases :

1) Recherche des tous les « patrons » ou itemsets fréquents, qui apparaissent dans la base de données avec une fréquence supérieure ou égale à un seuil défini par l'utilisateur, appelé minsup.
2) Génération, à partir de ces patrons fréquents, de l'ensemble des règles d'association ayant une mesure de confiance supérieure ou égale à un seuil défini par l'utilisateur, appelé minconf.

4.2. L'algorithme K-PPVBA-2T

K-PPVBA est une amélioration de l'algorithme des K-PPV. L'objectif est d'attribuer des poids à chaque attribut en utilisant les règles d'association. Nous avons utilisé les règles d'associations qui permettent d'identifier les termes les plus représentatifs d'une classe donnée. Chaque transaction est composée de l'ensemble des termes extraits (attributs) et de l'étiquette de la classe. Après la génération des règles, on ne garde que les règles de type :

$$Attribut \rightarrow Classe \quad \text{et} \quad Attribut_1, Attribut_2 \rightarrow Classe.$$

Les règles composées de trois attributs sont rares et ne sont pas déterminantes.
L'idée est que si deux attributs, $attribut_1$ et $attribut_2$ sont associés ensembles à une classe et si chacun d'eux individuellement est associé à la même classe, ces deux attributs sont jugés pertinents. La force informationnelle de chacun des deux attributs déduite de leur association, est plus importante que la force informationnelle d'un attribut seul.
Nous l'appliquons en deux versions : une première (K-PPVBA-1T) où nous ne prenons en compte que les règles composées d'un seul attribut (Terme). Une deuxième version (K-PPVBA-2T) où les règles d'un seul attribut sont déduites des règles de deux attributs.
La fonction de pondération se base sur deux paramètres : le plus grand support pour chaque attribut noté G_sup et aussi la plus grande confiance pour chaque attribut appelé G_conf. Par conséquent, la formule de la distance de l'algorithme K-PPV doit être modifiée en ajoutant le vecteur poids (W) défini comme :

$$W[i] = \left(\frac{1}{1 - G\sup_{[i]}}\right)$$

La nouvelle formule de calcul de distance utilisée dans la méthode K-PPVBA-2T, s'écrit alors : $D(a,b) = \sqrt{\sum_{i=1}^{n} W[i] \times (x_{ai} - x_{bi})^2}$

où a et b sont deux documents, x_{ai} et x_{bi} représentent le terme i de chaque vecteur document.

Le processus général de l'approche K-PPVBA-2T est décrit dans la figure 5 et est composé de trois phases :

Phase 1 : cette phase est constituée de deux étapes. La première étape est la construction des transactions qui représenteront les entrées pour générer les règles d'associations. Chaque document est transformé en une transaction, constituée de l'ensemble des descripteurs représentatifs du document associée à l'étiquette de la classe. La deuxième étape est la génération des règles d'association grâce à un algorithme de recherche de type Apriori [1].
Phase 2 : dans cette phase, nous cherchons à générer un vecteur poids pour tous les attributs de l'espace de description des documents. Pour chaque attribut, un groupe de 15 règles (15 correspondant au nombre de classes) est construit. La règle la plus pertinente (le support le plus élevé, la confiance la plus élevée) est retenue. Le vecteur poids est construit d'après la formule indiquée dans l'algorithme.
Phase 3 : cette phase permet d'appliquer l'algorithme K-PPV avec l'extension ajoutée. Pour prédire la classe d'un nouveau document par le calcul de la similarité inter-document, nous prenons en compte le vecteur poids généré dans la phase précédente.

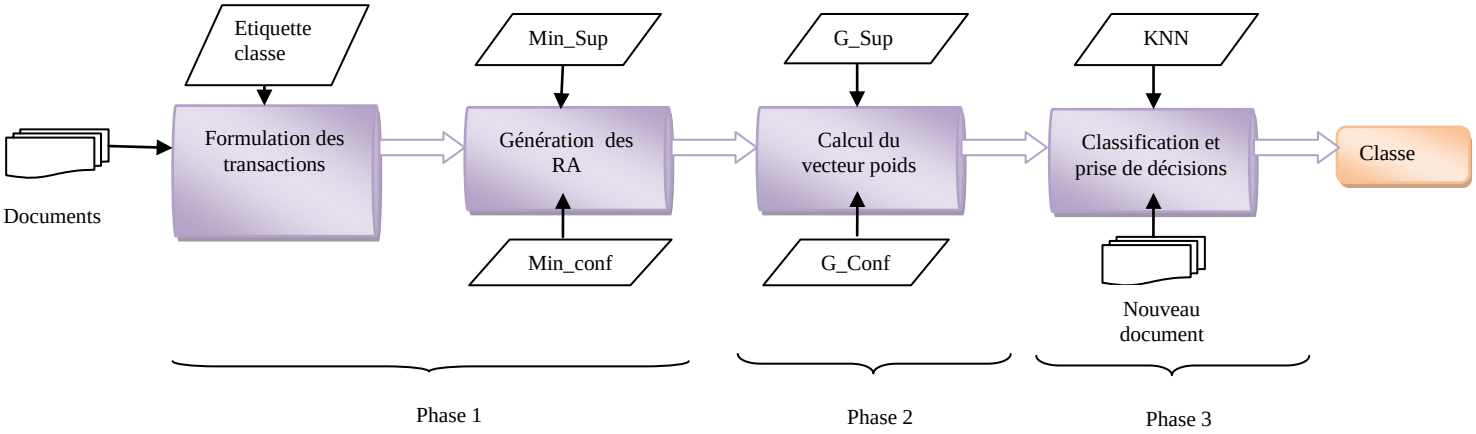

Figure 5 : Processus générale de l'approche K-PPVBA

Cette technique étend ainsi la méthode des K-plus-proches-voisins selon deux voies :

1) Tout d'abord, un schéma de pondération des descripteurs est introduit en fonction de leur poids informationnel par rapport à toutes les classes.
2) Le vote des plus proches voisins est basé sur une fonction étendue par le vecteur w. La seconde extension utilise la force d'activation des termes par rapport à toute la distribution des classes.
Cette extension est fondée sur l'idée que les observations de l'échantillon d'apprentissage, qui sont particulièrement proches de la nouvelle observation (y, x), doivent avoir un poids plus élevé dans la décision que les voisins qui sont plus éloignés du couple (y, x). Ce n'est pas le cas avec la méthode K-PPV : en effet seuls les k plus proches voisins influencent la prédiction, mais l'influence est identique pour chacun des voisins, indépendamment de leur degré de similarité avec (y, x). Pour atteindre ce but, les distances, sur lesquelles la recherche des voisins est fondée dans une première étape, sont transformées en fonction de la force (i.e. du pouvoir) du terme à activer la classe.

Tableau 3 : Comparaison de résultats de la classification avec K-PPV et K-PPV-BA

	K-PPV	*K-PPV-BA-1T*	*K-PPV-BA-2T*
Précision	*0,61*	*0,65*	*0,67*

Le tableau 3 donne les résultats de précision obtenus après l'application des trois algorithmes d'apprentissage sur le corpus des notices (lemmes : NVA) avec une pondération TF-IDF. Les meilleurs résultats sont obtenus avec l'algorithme K-PPVBA-2T comparativement aux algorithmes K-PPV et K-PPV -1T. Nous constatons que le pourcentage des notices bien classées, grâce à la pondération du vecteur W, passe de 61% à 65% avec K-PPVBA-1T et à 67% avec K-PPVBA-2T. L'approche K-PPV BA-2T améliore donc sensiblement les performances de la classification sur notre base de test.

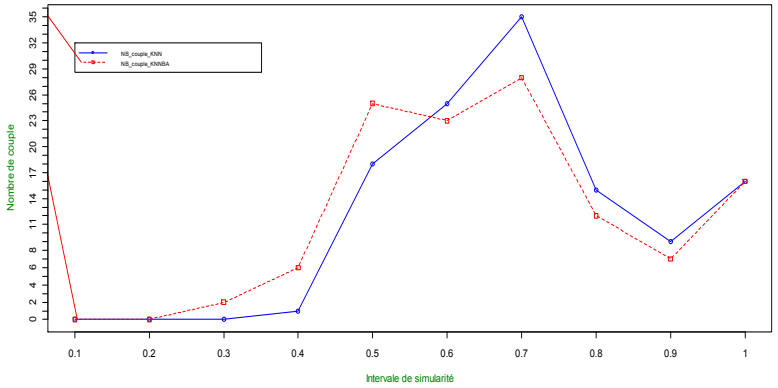

Figure 6 : Correction du déséquilibre et de la similarité des classes avec l'approche K-PPVBA-2T

Comme le montre la figure 6, nous avons, grâce à cette approche, joué à la fois sur la correction de la distribution des termes dans les classes et sur la correction de la similarité entre les classes. Comme le montre également la figure, le lissage de la distribution des termes n'est cependant pas effectif sur la plus grosse classe (A61K31) qui reste toujours une classe majoritaire.

5. Discussion et Conclusion

La classification d'articles scientifiques dans un plan de classement de brevets est un véritable challenge, ce type de plan étant très détaillé et finalement peu adapté contenu des documents scientifiques.

Dans cet article nous avons présenté une nouvelle méthode de classification supervisée issue de la méthode des K-PPV. Cette méthode que nous avons nommée K-PPV-BA-xT exploite une pondération des termes descripteurs des classes basée sur les règles d'association induites par ces termes. Nous l'avons appliqué sur un corpus de notices bibliographiques issues de la base Medline, dans le but des les classer dans un plan de classement de brevets du domaine de la pharmacologie. Cette nouvelle méthode offre des performances très intéressantes dans notre cas d'étude. Cependant le déséquilibre et la similarité de la description des classes obtenues restent toujours des problèmes majeurs qui freinent l'amélioration des performances de la classification automatique des

notices dans le plan internationale des brevets. C'est pourquoi, nous avons entrepris de nouvelles expérimentations dans le but de combiner notre méthode à des techniques alternatives, et notamment à des techniques plus spécifiques de gestion de l'équilibre des classes.

Nous comptons également expérimenter notre méthode K-PPVBA-2T avec des benchmarks internationaux reconnus afin de nous comparer plus efficacement aux algorithmes existants.

Remerciements : ce travail a été réalisé dans le cadre du programme QUAERO[5] financé par OSEO[6], agence nationale de valorisation de la recherche.

6. Bibliographie

[1] AGRAWAL, R. et SRIKANT R. Fast algorithms for mining association rules in large data bases. Journal of Computer Science and Technology (1994) Volume: 15, Issue: 6, Publisher: Morgan Kaufmann Publishers Inc., pp. 487-499.

[2] APTE, C., DAMERAU, F. et WEISS S. M. Text mining with decision rules and decision trees. Proceedings of the Conference on Automated Learning and Discovery, Workshop 6: Learning from Text and the Web, 1998.

[3] COHEN, A.M. et HERSH, W.R.: *A survey of current work in biomedical text mining*. Briefings in Bioinformatics 6, pp. 57-71, 2005.

[4] CORMACK, G. V. et LYNAM, T. R. *Online supervised spam filter evaluation*. ACM Transactions on Information Systems, 25(3):11, 2007.

[5] DURANT, K. et SMITH, M. *Predicting the Political Sentiment of Web Log Posts Using Supervised Machine Learning Techniques Coupled with Feature Selection*. In Advances in Web Mining and Web Usage Analysis: 8th International Workshop on Knowledge Discovery on the Web, Webkdd 2006, pages 187–206, Philadelphia. Springer-Verlag New York Inc,, 2007.

[6] EVANS, M., MCINTOSH, W., LIN, J., et CATES, C. *Recounting the courts? Applying automated content analysis to enhance empirical legal research*. Journal of Empirical Legal Studies, 4(4):1007–1039, 2007.

[7] FONTAINE, JF., BARBOSA-SILVA, A., SCHEFER, M., HUSKA MR., MURO EM. et ANDRADE-NAVARRO, MA. MedlineRanker: flexible ranking of biomedical literature. Nucleic Acids Res 37(Web Server issue): W141-W146, 2009.

[8] HILLARD, D., PURPURA, S., et WILKERSON, J. *An active learning framework for classifying political text*. In Annual Meeting of the Midwest Political Science Association, Chicago (2007).

[9] IYER, R.; LEWIS, D.; SCHAPIRE, R.; SINGER, Y.; et SINGHAL, A. Boosting for document routing. In Proceedings of the Ninth International Conference on Information and Knowledge Management, 2000.

[10] JOACHIMS, T.: A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization in Proceedings of ICML-97, 14th International Conference on Machine Learning, 1997

[11] 16 JOACHIMS T., «Text categorization with support vector machines: Learning with many relevant features». In proceedings of the European conference on Machine learning, pp. 137-142, 1998.

[5] http://www.quaero.org
[6] http://www.oseo.fr/

[12] JONES, S. et KAREN. A statistical interpretation of term specificity and its application in retrieval, Journal of Documentation, pp. 11-21, 1972.

[13] KRALLINGER, M., VAZQUEZ, M, LEITNER, F, SALGADO, D et VALENCIA, A. Results of the BioCreative III (Interaction) Article Classification Task. In Proceedings of the Third BioCreative Workshop, Bethesda, USA, 13-15 September 2010, 2010

[14] LAN, M., TAN, C.L., SU, J. et LOW, H.B.: Text representations for text categorization: a case study in biomedical domain. In: IJCNN: International Joint Conference on Neural Networks. 2007.

[15] LEWIS D. D. et RINGUETTE, M., «*Comparison of two learning algorithms for text categorization*», In Proceedings of the 3rd Annual Symposium on Document Analysis and Information Retrieval (SDAIR'94), pp. 81-93, 1994.

[16] LEWIS, D. D., «*An Evaluation of Phrasal and Clustered Representations on a Text Categorization Task*», ACM 15th Ann Int'l SIGIR'92, 1992, pp. 37-50, 1992.

[17] MORDIAN, M. et BAARANI, A. KNNBA: k-Nearest Neighbours Based Association Algorithm. University of Isfahan, Iran, 2009.

[18] PANG, B. et LEE, L. *Opinion mining and sentiment analysis*. Foundations and Trends in Information Retrieval, 2(1-2):1–135, 2008.

[19] PURPURA, S. et HILLARD, D. *Automated classification of congressional legislation*. Proceedings of the international conference on Digital government research, pages 219–225, 2006.

[20] SALTON G. et BUCKLEY C., Term-weighting approaches in automatic text retrieval, Information Processing Management, pp. 513-523, 1988.

[21] SALTON, G., Automatic processing of foreign language documents. Prentice-Hall, Englewood Cliffs, Nj, 1971.

[22] SCHAPIRE, R.; SINGER, Y.; et SINGHAL, A. Boosting and Rocchio applied to text filtering. In Proceedings of the 21st Annual International Conference on Research and Development in Information Retrieval, 1998.

[23] SCHÜTZE, H., HULL, D. A et PEDERSEN, J. O. *A Comparison of Classifiers and Document Representations for the Routing Problem*. Proceedings of the 18th Annual ACM SIGIR Conference, pp. 229--337, 1995.

[24] SCHMID H. Probabilistic part-of-speech tagging using decision trees. In Proceedings of the International Conference on New Methods in Language Processing, pp. 44–49, 1994.

[25] SEBASTIANI, F., A tutorial on automated text categorisation, In Analia Amandi and Ricardo Zunino, editors, Proceedings of the 1st Argentinian Symposium on Artificial Intelligence (ASAI'99) pp. 7-35, 1999.

[26] SUOMELA, BP. et ANDRADE, MA. Ranking the whole MEDLINE database according to a large training set using text indexing. BMC Bioinformatics 6:75, 2005.

[27] QUNINLAN, J.R., «*Induction of decision trees*», Machine Learning, 1(1), pp. 81-106, 1986.

[28] VINCARELLI A., Indexation de documents manuscrits, In Proceedings du Colloque International Francophone sur l'Ecrit et le Document (CIFED06), pp. 49-53, 2006.

[29] WIENER, E., PEDERSEN, J. O. et WEIGEND, A. S.. *A Neural Network Approach to Topic Spotting*. Symposium on document analysis and information retrieval, pp. 317-332, 1995.

[30] YANG, Y. et CHUTE, C.G. *An example based mapping method for text categorization and retrieval*. ACM Trans. Inform. Syst., 12: 252-277, 1994.

[31] YANG Y. et LIU X., A reexamination of text categorization methods, In SIGIR, ACM, pp. 42-49, 1999.

[32] YIN L, XU G, TOTII M, NIU Z, MAISOG, JM., WU C, HU Z et, LIU H. Document classification for mining host pathogen protein-protein interactions. Artif. Intell. Med 49(3):155-160, 2010.

1. Annexe

Algorithme: KNNBA-2T

Entrées :

D: est un modèle des données d'apprentissage étiquetées avec Ri ;
Ri : une instance de D tqri : (Att1i,Att2i,…,Attni,ClassEtq) ;
T: est un nouveau Modèle de test non étiquetées ;
Tab_G_Sup: Tableau de supports de tous les attributs ;
Tab_G_Conf: Tableau de confiance de tous les attributs ;
Min_Sup : seuil de support défini par l'utilisateur ;
Min_Conf: seuil de confidence défini par l'utilisateur ;

Sortie :

Etiquète de T;

Méthode :

For j=1 to n
{
if(Tab_G_Sup[j] <= Min_Sup OR Tab_G_Conf[j] <=Min_Conf)

W[j] = 0;

else

$$W[j] = (\frac{1}{1 - G_{\sup[j]}});$$

}

For each (Ri in D)
{
$$Dist = 0;$$
For j=1 to n
$$Dist = Dist + W[i] + (Att_i - Att_j)^2;$$
$$Dist = \sqrt{Dist;}$$
}
Choix de k: k plus proches voisins déterminé par l'utilisateur (impair);
Retour l'étiquette de classe ;

End : fin classification ;